Optimal Quality of Service Control in Communication Systems

Bosman, J.W.

2014

document version
Publisher's PDF, also known as Version of record

Summary

Optimal QoS control in Communication Systems

In current practice, quality of composite services is usually controlled on an ad-hoc basis, while the consequences of failures in service chains are often not well understood. A main concern is that, although such an approach might work for small chains, it will become unfeasible for future complex global-scale service chains. This raises the need for mechanisms that enable efficient usage of available shared resources while preserving the desired Quality of Service (QoS) as perceived by the end user. There are many optimization mechanisms available that could accomplish this. The problem is that in general these mechanisms are not suitably tailored for the current and evolving information and communication systems. The controls and thresholds are often based on simple improvised rules. As a consequence, the enormous potential of QoS mechanisms to enhance service quality remains largely unexploited.

The main challenge that is faced in this dissertation is: *how to effectively use QoS mechanisms for large-scale complex ICT systems with shared resources.*

To this end, we develop, analyze, optimize and evaluate quantitative models that capture the dynamics of QoS-control mechanisms and their implications on the user-perceived QoS. The development of efficient QoS mechanisms is complicated by the omnipresence of the phenomenon of uncertainty. Stochastic models are instrumental to capture such uncertainties and provide a basis for educated control of systems with uncertainty. One may distinguish the following three types of uncertainty.

Uncertainty about demand for resources. An inportant deal of demand for resources is driven by predictable user behavior. However, there are also many factors that are inherently unpredictable but may have a huge impact on resource availability (cyber attacks, flash crowds). For this purpose, mechanisms are required that can respond to this unpredictable behavior and provide robustness to threats and undesired behavior.

Variability in resource availability (shared resources). Various factors contribute to variability in resource availability such as resource sharing, network or system failure, chaotic behavior, and temporary overload. For a majority of Internet resources, capacity is shared among the different users. As a result, in the perspective of the users, the availability of resource capacity varies. Another contributing factor to variablity that may need explanation here is chaotic behavior. Chaotic behavior may for

example be caused by unexpected interactions between systems, often due to misconfiguration. In worst cases misconfiguration causes network or system failures. This is especially the case for (global) systems where demand volumes are so high that individual systems cannot handle all demand.

Limited information. Many existing models assume that the stochastic behavior of demand and resources is known. In practice, however this is rarely the case. Typically external parties at best have limited information about the internal behavior of a system. Also external factors impact the challenge of limited information from system behavior. Systems possibly operate in changing environments driven by uncertain, unpredictable factors. To respond in a fashionable way, mechanisms are required that can adapt to these changes.

Over the past few years, the tremendous popularity of smart mobile end devices and services (like YouTube) has boosted the demand for streaming media applications offered via the Internet. As the Internet provides no more than best-effort service quality, packet streams generated by streaming media applications are distorted by fluctuations in the available bandwidth, which may be significant over the duration of a typical streaming application. To cope with these distortions, play-out buffers temporarily store packets so as to reproduce the signal with a fixed delay offset. In Chapters 2 and 3 we study a video stream model where the network is modeled as a Markov Modulated fluid queue. In this model a Contineous Time Markov Chain represents the actual transmission rate through the network. Chapter 2 considers a two-state transmission rate model while Chapter 3 considers a more general transmission rate model. For the play-out buffer an initial buffer level b_{init} is determined such that the probability that the video will stall during play-out will not exceed an agreed service level probability p_{empty}. We show that the probability of this event corresponds to the probability of the event where the maximum congestion level $M(t)$ exceeds the initial buffer level b_{init}. From this insight we derive an expression that maps p_{empty}, T_{play} and the network and video parameters to a minimal buffer level b_{init}. Simulation results indicate that the buffer level that is obtained from our analysis is a conservative estimate, i.e., it overestimates the true minimal required buffer level.

In Chapter 4 we consider the transmission of file flows across multiple parallel wireless networks. Each wireless network is modeled as a processor sharing node. In this setting background flows are generated by clients with only one available network connection while foreground flows are generated by clients with multiple network connections. The goal is to minimize the expected transfer time of elastic data traffic by smartly dispatching the jobs of foreground flows to the networks. However only partial information is available in the sense that only the sum of the numbers of foreground and background flows can be observed. To this end, we propose a simple index rule called the convex combination (CC) rule. Extensive simulations with real networks show that this method performs extremely well under practical cir-

cumstances for a wide range of realistic parameter settings. The method presented in this chapter is a simple index rule that is essentially a convex combination of techniques that are found to work well extreme cases. To assess the effectiveness of the CC method, we have performed extensive simulation experiments in a real network simulator that implements the full wireless protocols stack. The results show that the CC method leads to close-to-optimal performance for a wide range of realistic parameter settings.

In Chapter 5 we investigate a general class of dynamic resource allocation problems that involve different types of resources and uncertain/variable demand. Aiming to maximize the expected net-benefit based on rewards and costs from the different resources, an optimal dynamic control policy has been derived within a singular stochastic optimal control setting. The mathematical analysis includes obtaining simple expressions that govern the dynamic adjustments to resource allocation capacities over time under the optimal control policy. Based on this analysis, a wide variety of extensive numerical experiments have been constructed. The results demonstrate and quantify significant benefits of the optimal dynamic control policy over recently proposed alternative optimization approaches in addressing a general class of resource allocation problems across a diverse range of application domains. Moreover, our results strongly suggest that the approach taken in this chapter can provide an effective means to develop easily-implementable online algorithms for solving stochastic optimization problems.

In Chapter 6 we address dynamic decision mechanisms for composite web services. We represent the composite web-service as a (sequential) workflow of tasks. For each task within this workflow, a number of third-party service alternatives may be available, offering the same functionality at different price-quality levels. Before a task in the workflow can be executed, a service alternative must be selected that implements the task functionality. We have developed a model to maximize benefit for composite services by on-the-fly dynamic service selection. The selection decisions are based on observed response times, the response-time characteristics of the alternative, the end-to-end response-time objectives, and the reward and penalty parameters. The results not only indicate *that* there is an enormous potential gain compared to other, non-dynamic approaches, but also show *how* one can realize such gains. We believe that this work is a significant step in realizing cost-efficient provisioning of complex composite services.

In Chapter 7 we propose a runtime closed-loop control mechanism that dynamically optimizes service composition in real time by learning and adapting to changes in third party service response time behaviors. We extend the dynamic programming approach of Chapter 6 to a closed-loop approach where dynamic programming is applied on empirical distributions resulting from the actual realized response-times of third party service providers. Our approach is robust to changes in the sense that it adapts to changes in response-time distributions of concrete service alternatives.

To achieve this we use a smoothing approach or a sliding window approach on the empirical distribution. The smoothing approach has the advantage that there is no overhead in bookkeeping of sliding window samples. When using our approach must strike a balance between parameters that we use in the optimization such as the sliding window W or exponential smoothing parameter κ and the change point detection test significance α. These parameter values are constrained by computational power and probe cost. Experimental results indicate that in an environment with changing response-time behavior our closed-loop approach has a significant advantage as it learns and exploits response-time behavior on the fly compared to a static lookup table that does not account for environment changes.